Journal of Informatics
Electrical and Electronics Engineering, 2020,
Vol. 01, Iss. 02, S. No. 004, pp. 1-15
ISSN (Online): 2582-7006

Neural network based direct MRAC technique for improving tracking performance for nonlinear pendulum system

Alemie Assefa

Department of Electrical and Computer Engineering, Debre Berhan University, Debere Berhan, Ethiopia
alemieastu@ gmail.com

How to cite this paper: A. Assefa (2020) Neural network based direct MRAC technique for improving tracking performance for nonlinear system. *Journal of Informatics Electrical and Electronics Engineering,* Vol. 01, Iss. 02, S. No. 4, pp. 1-15, 2020.

Received: 31/10/2020
Accepted: 20/12/2020
Published: 25/12/2020

Abstract

This paper investigates the application of a neural network-based model reference adaptive intelligent controller for controlling the nonlinear systems. The idea is to control the plant by minimizing the tracking error between the desired reference model and the nonlinear system using conventional model reference adaptive controller by estimating the adaptation law using a multilayer backpropagation neural network. In the conventional model reference adaptive controller block, the controller is designed to realize the plant output converges to reference model output based on the plant, which is linear. This controller is effective for controlling the linear plant with unknown parameters. However, controlling of a nonlinear system using MRAC in real-time is difficult. The Neural Network is used to compensate the nonlinearity and disturbance of the nonlinear pendulum that is not taken into consideration in the conventional MRAC therefore, the proposed paper can significantly improve the system behaviour and force the system to behave the reference model and reduce the error between the model and the plant output. Adaptive law using Lyapunov stability criteria for updating the controller parameters online has been formulated. The behaviour of the proposed control scheme is verified by developing the simulation results for a simple pendulum. It is shown that the proposed neural network-based Direct MRAC has small rising time, steady-state error and settling time for a different disturbance than Conventional Direct MRAC adaptive control.

Keywords

Model reference adaptive control (MRAC), Neural Network (NN), Multilayer Back propagation Neural Network.

1. Introduction

In the adaptive control, controlling of the nonlinear system with present-day sophistication and complexities has often been an important research area due to the difficulty in modelling, nonlinearities, and uncertainties. Model reference adaptive control is the best scheme used in the adaptive control technique. Recently MRAC has received considerable attention and many new approaches have been applied to the practical process [2]. In the MRAC scheme, the controller is designed to realize the plant output converges to reference model output based on assumption that plant can be linearized [3], [4], and [5]. Therefore, direct MRAC is best controller for controlling linear plants with unknown parameters. However, it may not guarantee for controlling nonlinear plants (Pendulum) with unknown structure. In recent years, an artificial neural network (ANN) has become very popular in many control applications due to their higher computation rate and ability to handle nonlinear systems [6].

The adaptive controller is designed to realize a plant output tracks to reference model output based on assumption that the plant can be linearized. [8], [9], [3] However, as most industrial processes are highly nonlinear, non-minimum, and with various type of uncertainties and load disturbances the performance of the linear MRAC may deteriorate, and suitable nonlinear control may have to be used.

In [11], the output of neural networks then adaptively adjusts the gain of the sliding mode controller so that the effects of system uncertainties eliminated and the output tracking error between the plant output and the desired reference signal can be asymptotically converging to zero. However, the sliding mode control action can lead to high frequency oscillations called chattering which may excite un-modeled dynamics, energy loss, and system instability and sometimes it may lead to plant damage.

2. Mathematical Modeling

Consider the simple pendulum shown in Figure 1, where l denotes the length of the rod and m denotes the mass of the bob. Assume the rod is rigid and has zero mass. Let θ denote the angle subtended by the rod and the vertical axis through the pivot point. The pendulum is free to swing in the vertical plane. The bob of the pendulum moves in a circle of radius l. To write the equation of motion of the pendulum, let us identify the forces acting on the bob. There is a downward gravitational force equal to mg, where g is acceleration due to gravity. There is also a frictional force resisting the motion (by air and any other frictions), which we assume to be proportional to the speed of the bob with a coefficient of friction b.

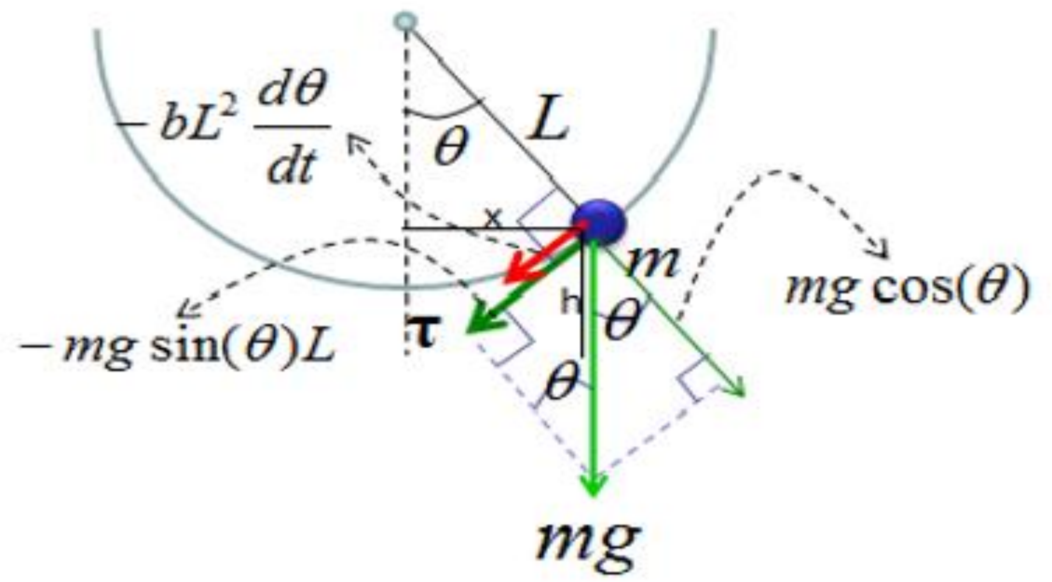

Figure 1 **Simple pendulum modeling**

$$x = l sin\theta, h = l(1 - cos\theta) \tag{2.1}$$

$$K.E = \frac{1}{2} I\omega^2 = \frac{1}{2} mv^2 \tag{2.2}$$

$$\omega = \frac{v}{l}, I = ml^2, v^2 = v_x^2 + v_y^2 \tag{2.3}$$

$$v_x = \frac{dx}{dt}, v_y = \frac{dy}{dt}, v_x^2 = \left(\frac{dx}{dt}\right)^2, v_y^2 = \left(\frac{dy}{dt}\right)^2 \tag{2.4}$$

Substituting equation (2.1) to equation (2.4)

$$v_x^2 = \left(\frac{dx}{dt}\right)^2 = \left(l\dot{\theta}cos\theta\right)^2, v_y^2 = \left(\frac{dy}{dt}\right)^2 = \left(l\dot{\theta}sin\theta\right)^2 \tag{2.5}$$

$$K.E = \frac{1}{2} m\left(v_x^2 + v_y^2\right) = \frac{1}{2} m(\left(l\dot{\theta}cos\theta\right)^2 + \left(l\dot{\theta}sin\theta\right)^2)$$

$$Therefore,$$

$$K.E = \frac{1}{2} m\left(\left(l\dot{\theta}\right)^2 [cos^2\theta + sin^2\theta]\right)$$

$$K.E = \frac{1}{2} m(\left(l\dot{\theta}\right)^2), since, cos^2\theta + sin^2\theta = 1 \tag{2.6}$$

$$P.E = mgh, but\ h = l(1 - cos\theta), P.E = mgl(1 - cos\theta) \tag{2.7}$$

By using Lagrangian equation

$$L = K.E - P.E = \frac{1}{2}m((l\dot{\theta})^2 - mgl(1 - cos\theta)) \tag{2.8}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) - \frac{\partial L}{\partial \theta} = \tau(torque) - b\dot{\theta} \tag{2.9}$$

$$\frac{\partial L}{\partial \dot{\theta}} = ml^2\dot{\theta}, \frac{\partial L}{\partial \theta} = mgl(-sin\theta) = -mglsin\theta, \tag{2.10}$$

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\theta}}\right) = ml^2\ddot{\theta}$$

$$Therefore, ml^2\ddot{\theta} + mglsin\theta + b\dot{\theta} = \tau(torque) \tag{2.11}$$

The mathematical model for a simple pendulum is that

$$ml^2\ddot{\theta} + mglsin\theta + b\dot{\theta} = \tau(torque) \tag{2.12}$$

The state space form of the pendulum is given by,
Let $x_1 = \theta, x_2 = \dot{\theta} = \omega$

$$\dot{x}_1 = 0x_1 + x_2 \tag{2.13}$$

$$\dot{x}_2 = -\frac{b}{ml^2}x_2 + \frac{1}{ml^2}[\tau - mglsinx_1]$$

On the other hand the state space form is given by

$$\left.\begin{array}{c} \dot{x}_1 = 0x_1 + x_2 \\ \dot{x_2} = \dfrac{1}{ml^2}[\tau - mglsinx_1 - bx_2] \end{array}\right\} \tag{2.14}$$

$\dot{x} = Ax + B\,\Lambda\,[u - \theta^T\varphi(x)]$, $\boldsymbol{\Lambda}$ control input uncertainty

3. Controller Design

3.1. Adaptive control

Adaptive control is the best control method used by a controller, which must adapt to a controlled system with parameters, which vary or are initially uncertain [12], [16]. For example, as an aircraft flies, its mass will slowly decrease because of fuel consumption; a control law is needed that adapts itself to such changing conditions.

3.1. 1 Model Reference Adaptive Control

MRAC is one of adaptive control classification technique as shown in figure 2 below. When designing a controller for a system, a control designer typically would like to know how the system behaves physically. This knowledge is usually captured in the form of a mathematical model.

There are generally two classes of adaptive control: direct adaptive control and indirect adaptive controller [1]. Direct adaptive controller methods adjust the control gains directly and indirect adaptive controller methods estimate unknown system parameters for use in the update of the control gains. Asymptotic tracking is the fundamental property of model-reference adaptive control, which guarantees that the tracking error tends to approximately zero in the limit.

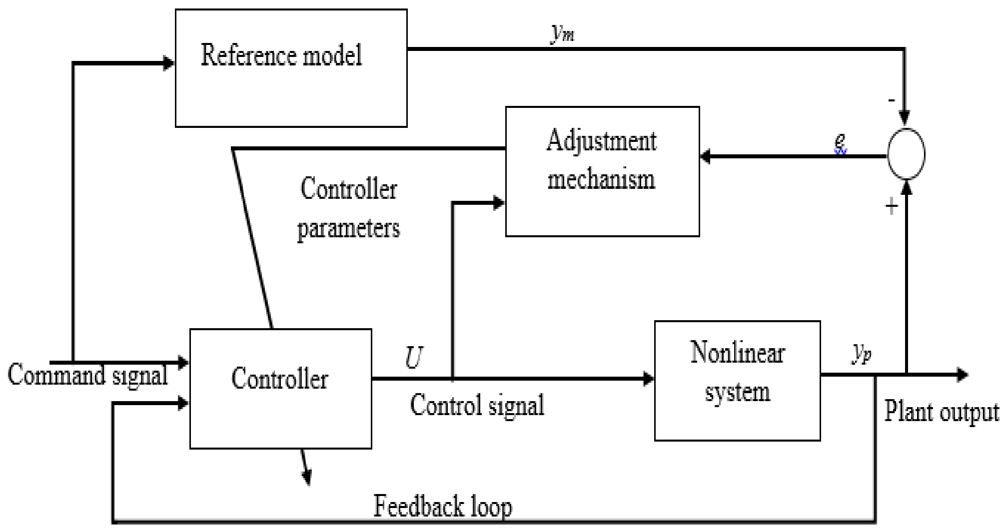

Figure 2 Block diagram of typical MRAC

3.2. Multilayer Neural Networks (Back propagation Algorithm)

The input signals are propagated in a forward direction on a layer-by-layer basis. Learning in a multilayer network proceeds the same way as for a perception. In a back-propagation neural network, the learning algorithm has two phases. First, a training input pattern is presented to the network input layer [19]. The network propagates the input pattern from layer to layer until the output layer generates the output pattern as shown in figure 3 below.

Second, if this pattern is different from the desired output, an error is calculated and then propagated backward through the network from the output layer to the input layer. The weights are modified as the error is propagated.

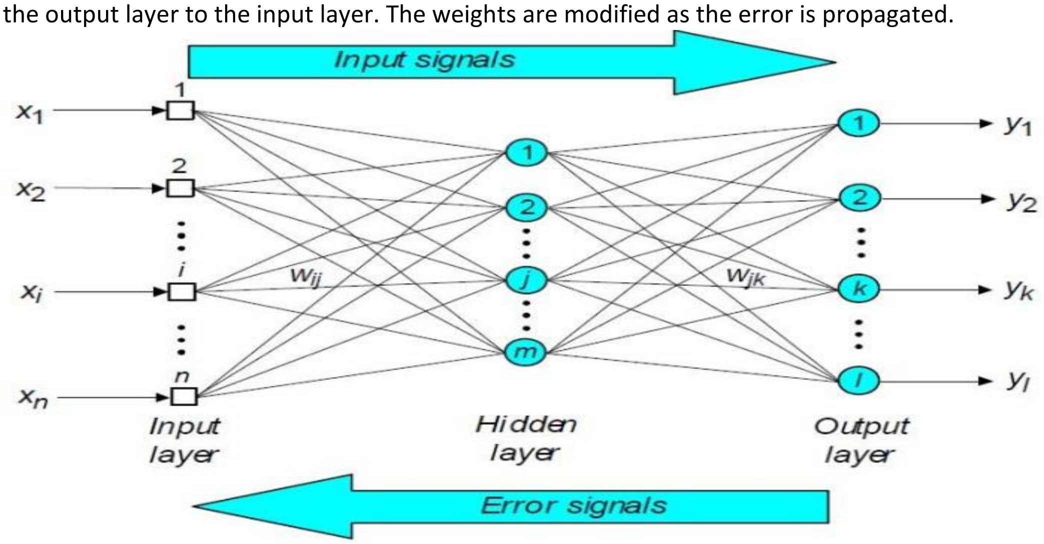

Figure 3 Back-propagation neural network

3.2.1. The Back propagation training algorithm

Back propagation is a common method for training a neural network, the goal of back propagation is to optimize the weights so that the neural network can learn how to correctly map arbitrary inputs to outputs. [19]

Back propagation method contains the following steps:

Step 1: Initialization; set all the weights and threshold levels of the network to random numbers uniformly distributed inside a range:

$$\left(-\frac{2.4}{Fi}, \quad +\frac{2.4}{Fi}\right) \tag{3.1}$$

Where Fi is the total number of inputs of neuron $\boldsymbol{i}$ in the network.

Step2: Activation; activate the back-propagation neural network by applying inputs $\mathrm{x1(p), x2(p), \ldots, xn(p)}$ and desired outputs $y_{d1}(\mathrm{p}), y_{d2}(\mathrm{p}), \ldots, y_{dn}(\mathrm{p})$ (forward pass).

Calculate the actual outputs of the neurons in the hidden layer:

$$y_j(p) = sigmoid\left[\sum_{i=1}^{n} x_i\,(p).\,w_{ij}(p) - \theta_j\right] \tag{3.2}$$

Where n is the number of inputs of neuron j in the hidden layer.

Calculate the actual outputs of the neurons in the output layer:

$$y_k(p) = sigmoid\left[\sum_{j=1}^{m} x_{jk}\,(p).\,w_{jk}(p) - \theta_k\right] \tag{3.3}$$

Where **m** is the number of inputs of neuron **k** in the output layer

Step 3: Weight training (back-propagate):-Update the weights in the back-propagation network propagating backward the errors associated with output neurons.

Calculate the error gradient for the neurons in the output layer:

$$\delta_k(p) = y_k(p).\,[1 - y_k(p)].\,e_k(p) \tag{3.4}$$

Where, $\mathrm{e_k(p) = y_{d,k}(p) - y_k(p)}$

Calculate the weight corrections:

$$\Delta w_{jk}(p) = \alpha.\,y_j(p).\,\delta_k(p) \tag{3.5}$$

Update the weight at the output neurons:

$$w_{jk}(p+1) = w_{jk}(p) + \Delta w_{jk}(p) \tag{3.6}$$

Calculate the error gradient for the neurons in the hidden layer

$$\delta_j(p) = y_j(p).\,[1 - y_j(p)].\sum_{k=1}^{l} \delta_k(p)\,w_{jk}(p) \tag{3.7}$$

Calculate the weight corrections:

$$w_{ij}(p) = \alpha.\,x_i(p).\,\delta_j(p) \tag{3.8}$$

Update the weights at the hidden neurons:

$$w_{ij}(p+1) = w_{ij}(p) + \Delta w_{ij}(p) \tag{3.9}$$

Step 4: Iteration:-Increase iteration **p** by one, go back to **Step 2** and repeat the process until the selected error criterion is satisfied.

3.4. Adaptation law design for simple pendulum

From state space equation form of the simple pendulum in the equation shown below.

$$\dot{x}_1 = 0x_1 + x_2 \tag{4.1}$$

$$\dot{x}_2 = 0x + \frac{1}{ml^2}[\tau - mglsinx_1 - bx_2]$$

We can express as a general expression of $\dot{x} = Ax + B\,\Lambda\,[u - \theta^T\varphi(x)]$ because the parameters of mgl and b are unknown or vary with time. Therefore, we can assign as the estimates of the parameter $\theta_1(theta1), \theta_2(theta2)$ respectively and the known basis function $sinx_1$ and x_2 to $\varphi_1(x), \varphi_2(x)$ respectively.

From the above

$$\dot{x}_1 = 0x_1 + x_2 \tag{4.2}$$

$$\dot{x}_2 = \frac{1}{ml^2}[\tau - [\theta_1(theta1) \quad \theta_2(theta2)]\begin{bmatrix}\varphi_1(x)\\ \varphi_2(x)\end{bmatrix}]$$

Let the adaptive controller

$$\boldsymbol{\tau} = -K_x{}^*.x + k_r(t).r(t) + \theta^T(t)\varphi(x), \tag{4.3}$$

$$where, \Delta k_r = k_r(t) - k_r{}^*, \Delta\theta = \theta\,(t) - \theta\,*$$

However, $K_x{}^*$ is the ideal controller gains obtained from Linear Quadratic Regulator.

Based on the Lyapunov adaptive law in chapter 3 we can get the estimates of

$$\Delta\dot{k}_r = \dot{k_r} = \gamma_r r e^T P sgnB$$

$$\Delta\dot{\theta} = \dot{\theta} = \Gamma_\theta \varphi(x) e^T P sgnB$$

The dimension of theta and phi as shown

$$\theta = \begin{bmatrix}\theta_1\\ \theta_2\end{bmatrix} = \begin{bmatrix}mgl\\ b\end{bmatrix}, \varphi(x) = \begin{bmatrix}\varphi_1(x)\\ \varphi_2(x)\end{bmatrix} = \begin{bmatrix}sinx_1\\ x_2\end{bmatrix}, e^T = [e_1 \quad e_2]$$

Where $e_1 = x_{1m} - x_1$ and $e_2 = x_{2m} - x_2$ and γ_r, Γ_θ are adaptation rates.

Table 1 Physical parameters of simple pendulum

Parameters	Value	Unit
m(pendulum mass)	0.5	kg
l(length of the rod)	1	m
b(damping coefficient)	0.5	Ns/m or kg/s
g(gravity constant)	9.8	***m/s²***

Linearizing the nonlinear pendulum at equilibrium point (0,0) to design the reference model.

Therefore, from table 1, we can determine the sate space parameters as shown below.

$$\dot{X} = AX + BU$$

$$\begin{bmatrix}\dot{\Delta x_1}\\ \dot{\Delta x_2}\end{bmatrix} = \begin{bmatrix}0 & 1\\ -9.8 & -1\end{bmatrix}\begin{bmatrix}\Delta x_1\\ \Delta x_2\end{bmatrix} + \begin{bmatrix}0\\ 1\end{bmatrix}u$$

$$Y = CX + DU \Leftrightarrow y = [1\;0]\begin{bmatrix}\Delta x_1\\ \Delta x_2\end{bmatrix}$$

The general block diagram for conventional direct model reference control in figure 4 below.

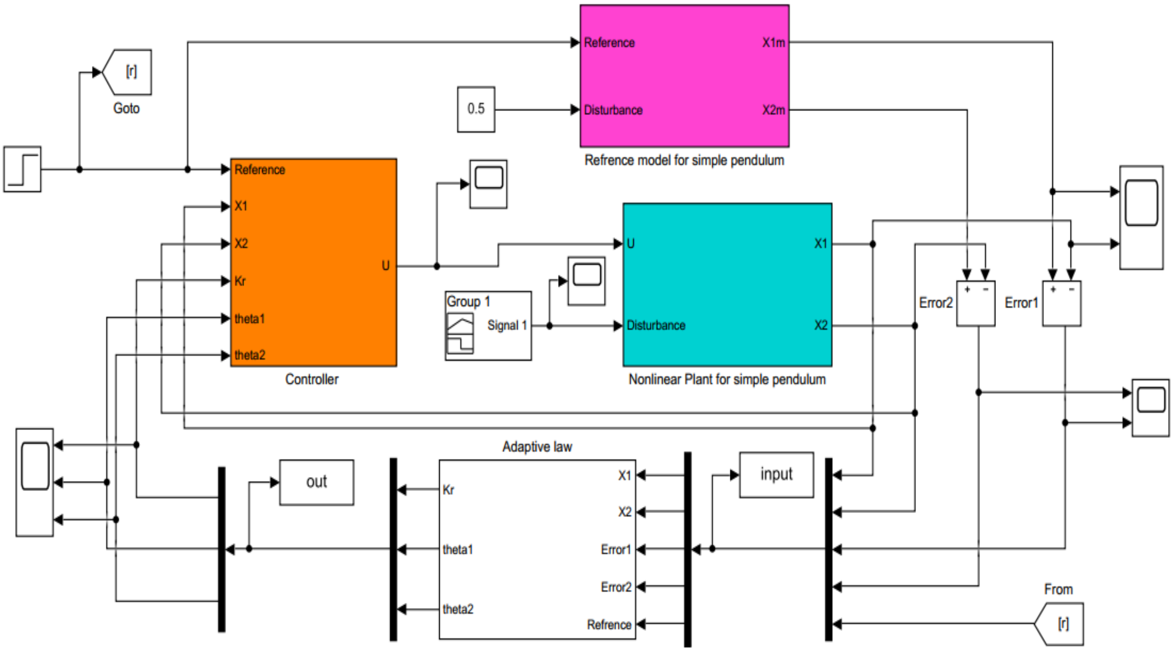

Figure 4 MATLAB/SIMULINK model for Conventional MRAC for nonlinear pendulum.

4. Simulation Results

4.1. Conventional Direct Model Reference Adaptive control for nonlinear pendulum result

The simulation result for conventional model reference adaptive control as shown below in figure 5 and figure 6.

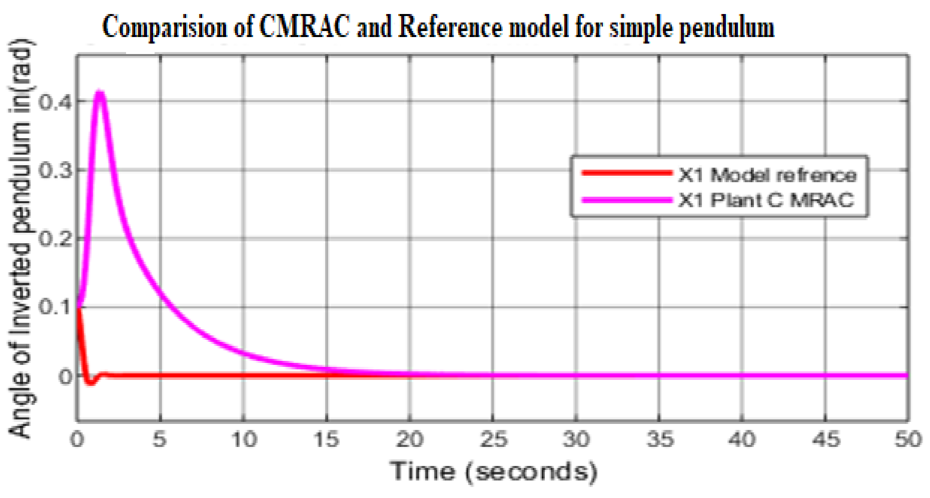

Figure 5 Angle response for IP disturbed from 0.1rad using Conventional MRAC

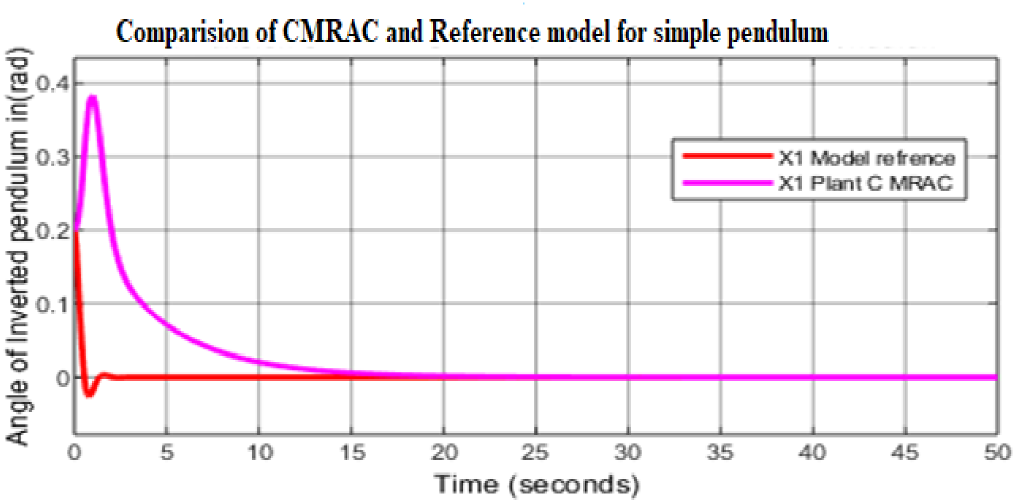

Figure 6 Angle response for IP disturbed from 0.2rad using Conventional MRAC

For a given stable equilibrium, 0deg input as reference the nonlinear system response tracks to the reference model after 15.5 seconds. In figure 7 below response for the nonlinear system the tracking error is high at the beginning so to reduce the tracking error Artificial neural network is preferable.

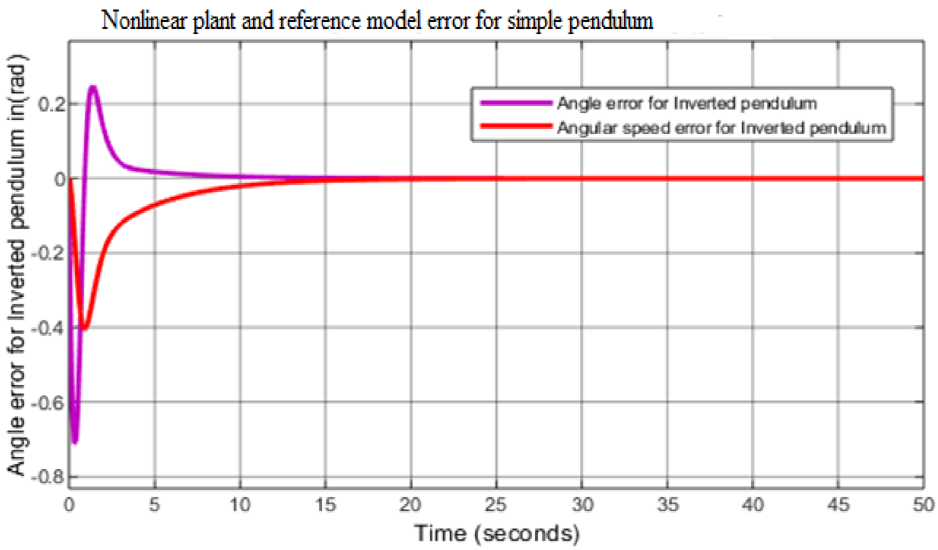

Figure 7 Angle Error between nonlinear plant and reference model to stable equilibrium point.

The parameter adjustment for conventional direct model reference adaptive control for nonlinear inverted pendulum as shown in figure 8 below for 0.1rad disturbance.

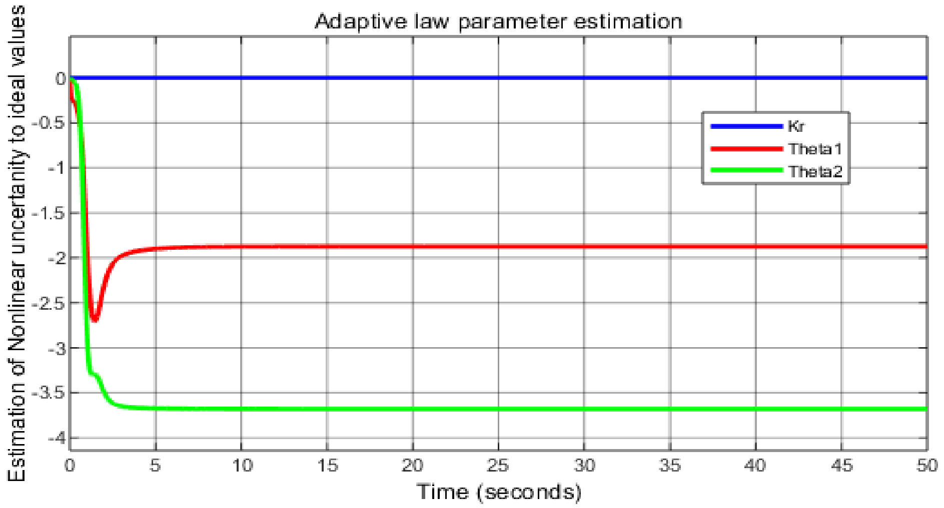

Figure 8 Estimated adaptive law result for 0.1rad disturbance for nonlinear pendulum

4.2. Neural network based direct MRAC for nonlinear pendulum result

In this paper, Artificial NN is used to improve the performance of nonlinear system by training of adaptation law to improve the plant due to rapidly estimating of the nonlinear uncertainty and unknown parameters to ideal values. Now NN is dividing into three layers, named the input layer with 5 neurons, the hidden layer with 20 neurons, and the output layer with 3 neurons. Backpropagation learning algorithm (Levenberg - Marquardt) is used to train the network. The ANN training data samples are collected from the input and output of Adaptation law. The hidden layer neurons are activated by using tan sigmoidal activation function and output a pure linear activation function is used.

The general block diagram for neural network based direct model reference adaptive control for nonlinear inverted pendulum shown in figure 9 below.

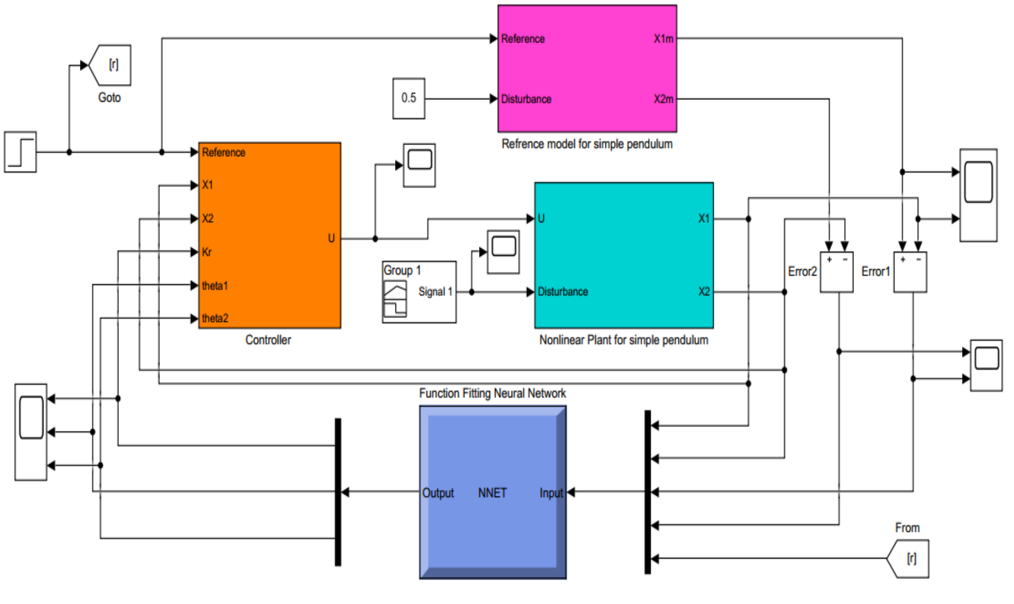

Figure 9 MATLAB/SIMULINK model for nonlinear pendulum using NN-MRAC

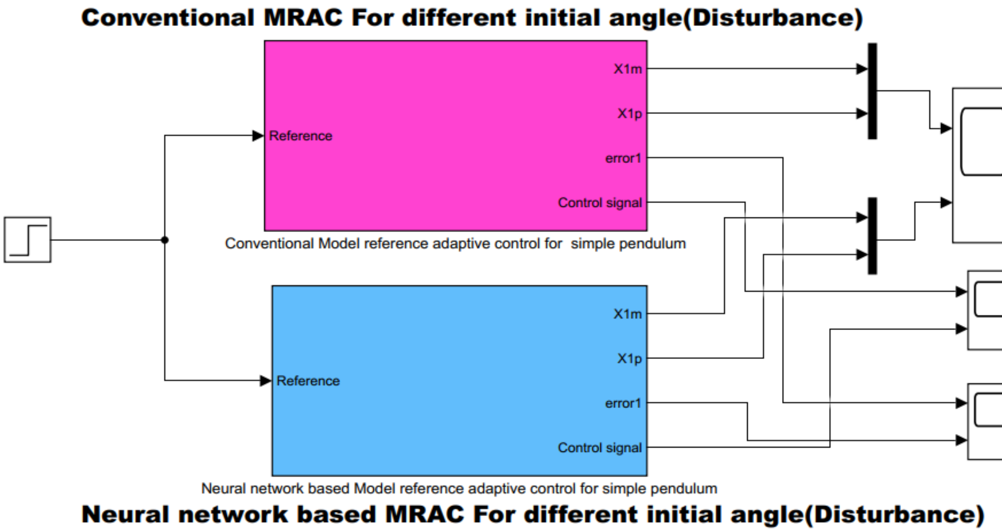

Figure 10 Comparison of both conventional MRAC and NN based MRAC for nonlinear pendulum

Using NN based MRAC and Conventional MRAC design the simulation result from figure 10 above block diagram for the simple pendulum system is illustrated in the figure (11-13) with different disturbances as shown.

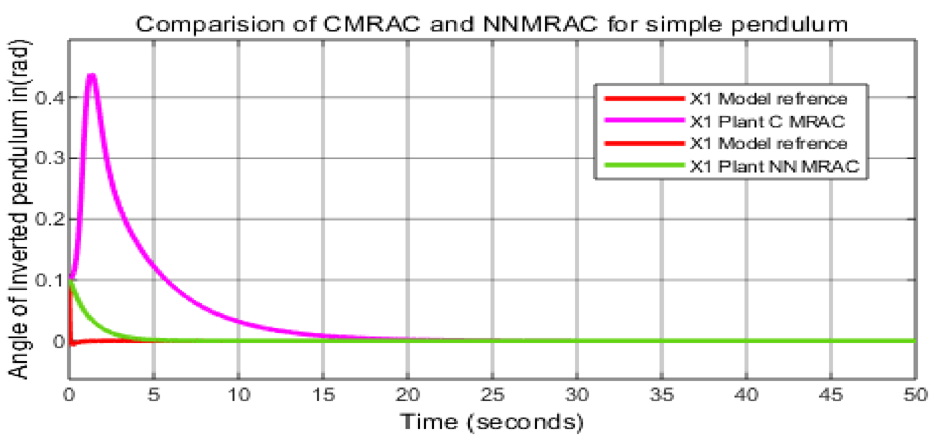

Figure 11 Comparison of Conventional MRAC and NN-MRAC angle response for IP disturbed from 0.1rad.

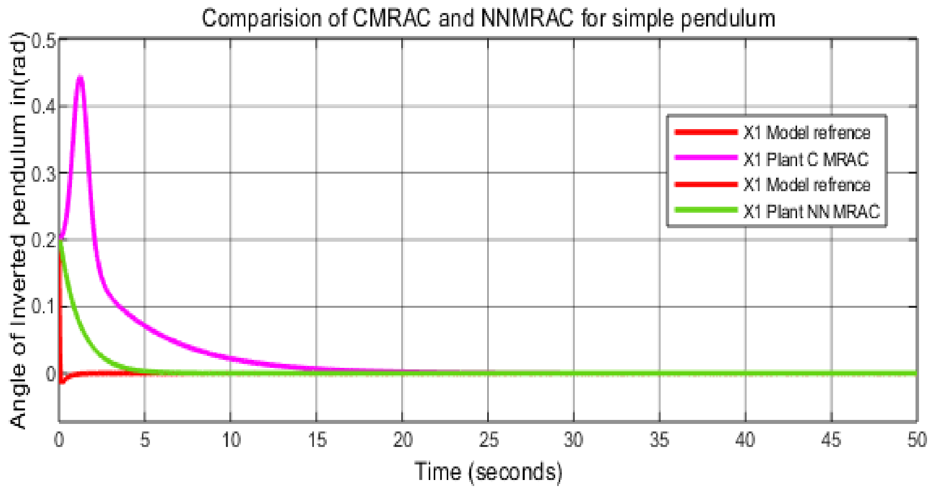

Figure 12 Comparison of Conventional MRAC and NN-MRAC angle response for pendulum disturbed from 0.2rad.

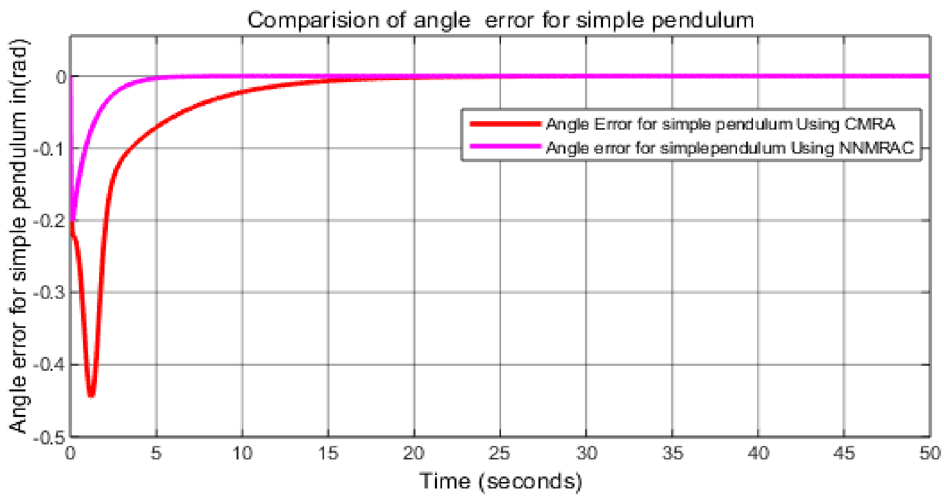

Figure 13 Comparison of angle error for pendulum given 0.2rad disturbance

From the above the NN based direct model reference control is better performance based on time specification i.e. rise time, settling time and steady state error than conventional direct MRAC for different disturbance. Generally given Disturbance as pulse input as shown figure 14 below and see the result both Conventional MRAC and NN based direct MRAC in figure (15-17) and the parameter adjustment for different disturbance in figure 18 shown below.

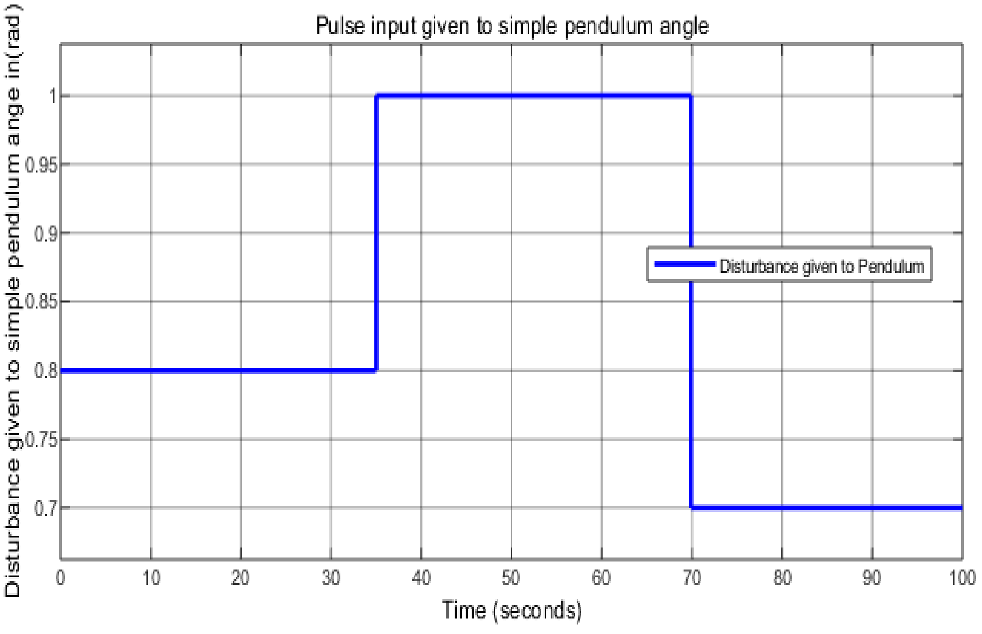

Figure 14 Different disturbance given to the angle of pendulum.

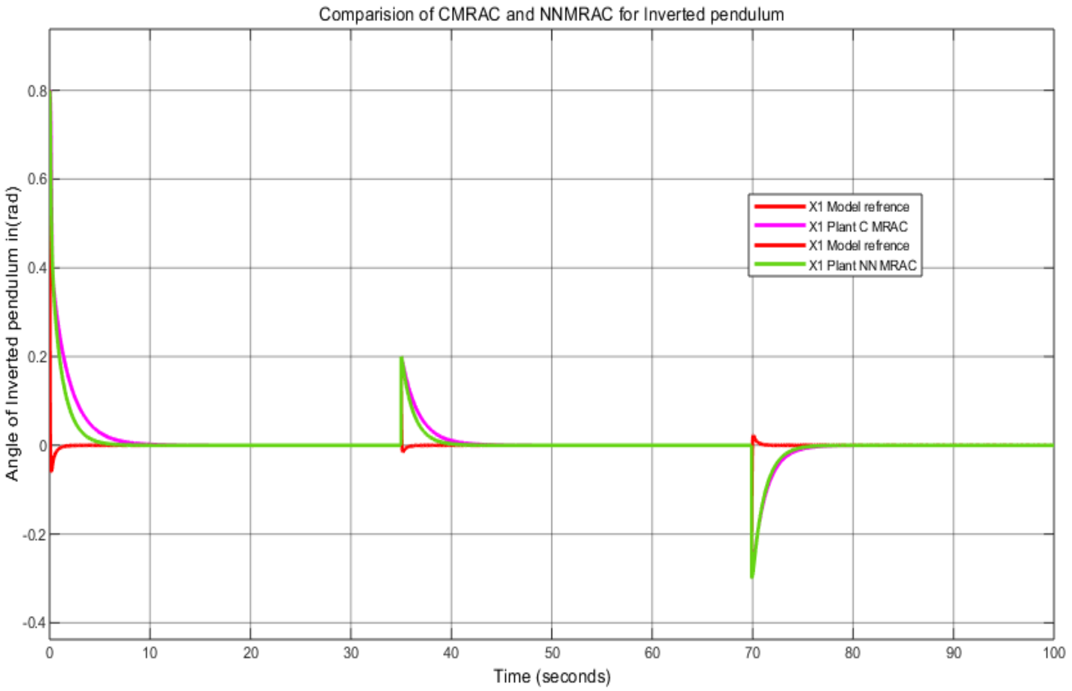

Figure 15 Comparison of angle response for Different disturbance given to angle of pendulum

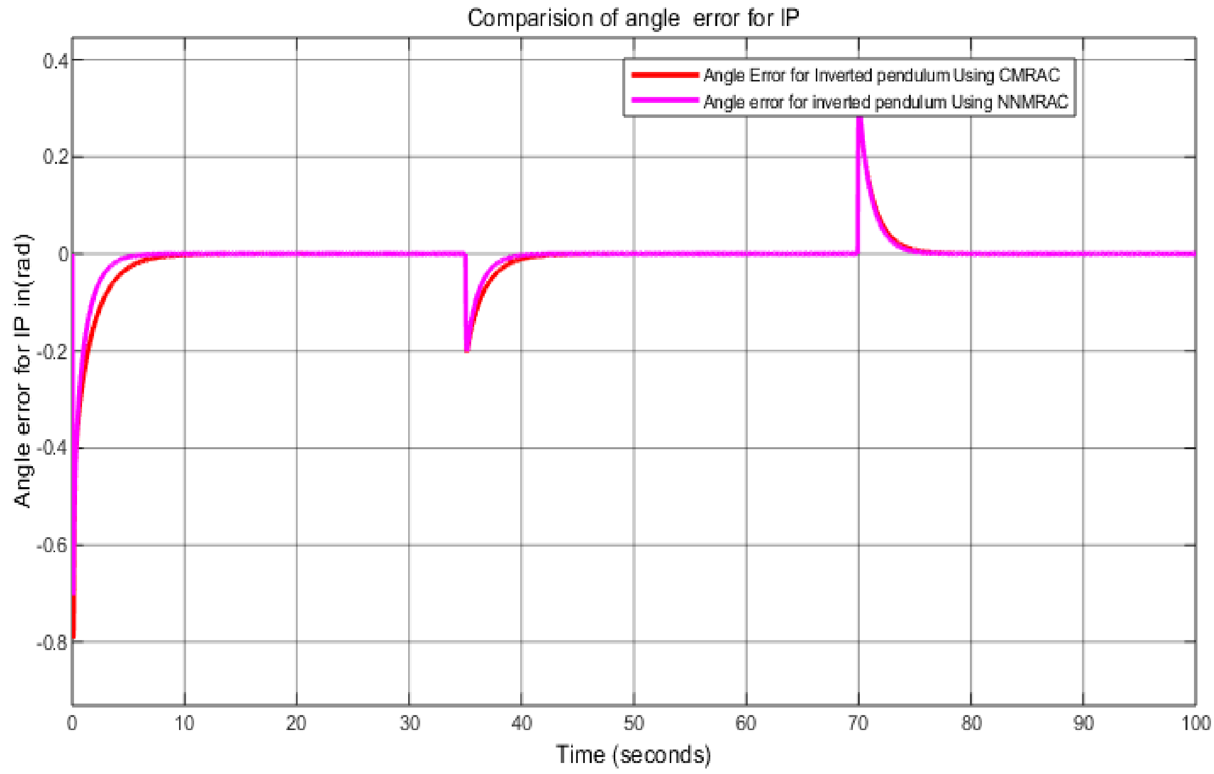

Figure 16 Comparison of angle error for a different disturbance on the angle of pendulum.

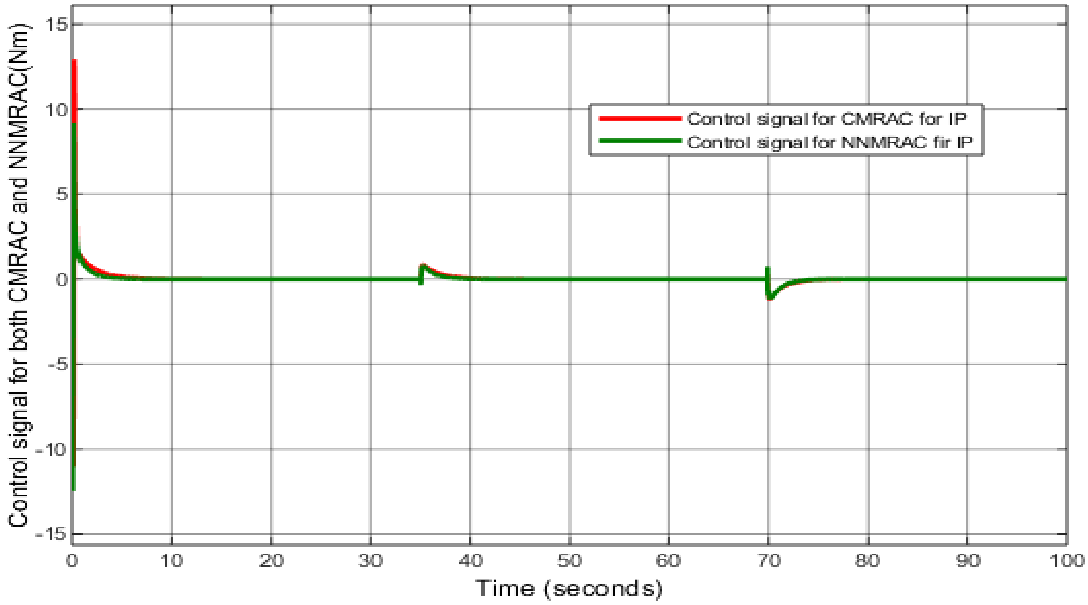

Figure 17 Control signal for simple pendulum for the different disturbance.

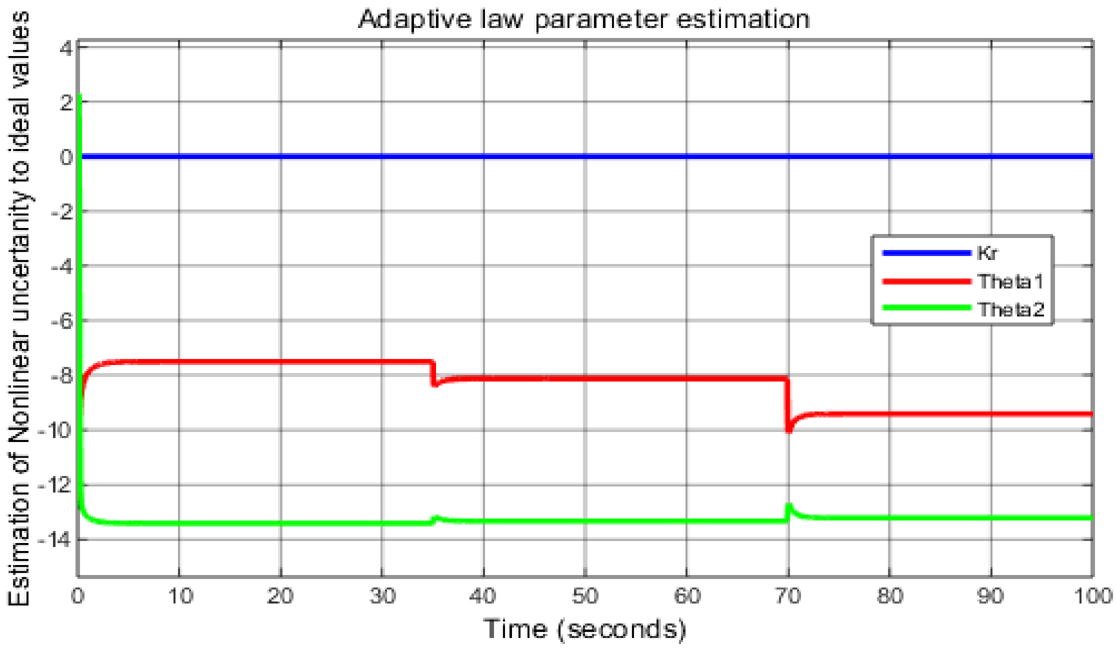

Figure 18 Estimated adaptive law result for different disturbance for pendulum

Finally, we can observe from the above graphs an ANN controller is a powerful controller to stabilize a nonlinear simple pendulum system when compared to Conventional MRAC. The conventional MRAC and an artificial neural network controller are designed for the stabilization of simple pendulum systems.

5. Conclusion

In this paper, modeling and designing of neural network based direct MRAC of nonlinear systems using MATLAB software to overcome the tracking performance of conventional direct model reference adaptive control to an equilibrium point for different disturbances have been investigated. Adaptive law using Lyapunov stability criteria for updating the controller parameters online has been formulated. Both the transient and steady state performances of the nonlinear system are improved by updating the parameters of neural networks (weight and biases) controller. The effectiveness of the proposed NN based DMRAC and Conventional MRAC is tested using MATLAB/SIMULINK. It is observed from the simulation results that the proposed NN based Direct MRAC has 3.13sec rise time, 5.15sec settling time for 0.1rad disturbance and 3.12sec rise time, 5.21sec settling time for 0.2rad disturbance. Whereas, the conventional direct MRAC has 5.42sec rise time, 15.5sec settling time for 0.1rad disturbance and 5.01sec rise time, 15.52sec settling time for 0.2rad disturbance. It is shown that the proposed neural network based Direct MRAC has smaller rising time, steady-state error and settling time for a different disturbance than Conventional DMRAC adaptive control.

Reference

[1] E. Laveretsky, K. A.Wise (2013) Robust and Adaptive control with Aerospace Application. Springer-Verlag London, pp. 317-353.

[2] N. T. Nguyen (2018) Model reference adaptive control. Springer international publishing, AG.

[3] K. B. Pathak, D. M. Adhyaru (2012) Survey of Model Reference Adaptive Control” IEEE International Conference, 06-08 Dec 2012.

[4] V.A. Arya, R.B. Aswin, A.E. George (2018) Modified Model Reference Adaptive Control for the Stabilization of Cart Inverted Pendulum System. International Research Journal of Engineering and Technology (IRJET), IRJET 5(4), 4592- 4596.

[5] R. J. Pawar, B. J. Parvat (2015) Design and implementation of MRAC and modified MRAC technique for the inverted pendulum. *Proceedings of the International Conference on Pervasive Computing (ICPC)*, Pune, India, 8-10 Jan. 2015, pp. 1-6.

[6] R. Prakash, R. Anita (2011) Design of Model Reference Adaptive Intelligent Controller Using Neural Network for Nonlinear Systems. *International Review of Automatic Control*, 4, no. 2, 153-161.

[7] M. Caudill (1987) Neural networks primer, part I. *AI Expert,* 2, no. 12, 46-52.

[8] Parks, P.C, "Lyapunov Redesign of Model reference adaptive control system", *IEEE Trans on Automatic Control, Vol.AC-11,* No.3 (1966). pp.362-367.

[9] Hang, Chang-Chieh, and P. C. Parks. "Comparative studies of model reference adaptive control systems." *IEEE transactions on automatic control* 18, no. 5 (1973): pp 419-428.

[10] Swarnkar, Pankaj, Shailendra Jain, and R. K. Nema. "Effect of adaptation gain in model reference adaptive controlled second order system." *Engineering, Technology & Applied Science Research* 1, no. 3 (2011): pp 70-75

[11] Zhihong, Man, X. H. Yu, K. Eshraghian, and M. Palaniswami. "A robust adaptive sliding mode tracking control using an RBF neural network for robotic manipulators." In *Proceedings of ICNN'95-International Conference on Neural Networks*, vol. 5, pp. 2403-2408. IEEE, 1995.

[12] Hunt, K. Jetal, D. Sbarbaro, R. Żbikowski, and Peter J. Gawthrop. "Neural networks for control systems—a survey."*Automatica* 28, no. 6 (1992): pp.1083-1112.

[13] Zhihong, Man, Hong Ren Wu, and Marimuthu Palaniswami (1998) An adaptive tracking controller using neural networks for a class of nonlinear systems." *IEEE transactions on neural networks* 9, no. 5, pp 947-955.

[14] *Munadi, M. Amirullah Akbar, To mohide Naniwa, Yoshiaki Taniai ," Model Reference Adaptive Control for DC Motor Based on Simulink",* 2016 6th International Annual Engineering Seminar (InAES).*IEEE Conferences,2016,pp.101-106*.

[15] K. Pirabakaran and V.M. Becerra," PID autotuning using neural networks and model reference adaptive control ", 15th Triennial World Congress Copyright © 2002 IFAC.

[16] Karl J. Astrom, Bjorn Wittenmark, "Adaptive Control" Pearson Education India, 2nd Edition, 2001.

[17] Adrian-Vasile Duka, Stelian Emilian Oltean, Mircea Dulau." Model Reference Adaptive Control and Fuzzy Model Reference Learning Control for the Inverted Pendulum. Comparative Analysis." *International Conference on dynamical systems and control, Venice, Italy, November 2-4*, 2005 (pp168-173).

[18] NemaR.K, Swarnkar Pankaj." Comparative Analysis of MIT Rule and Lyapunov Rule in Model Reference Adaptive Control Scheme" Innovative Systems Design and Engineering. Vol 2, No 4, 2011.

[19] Dr. Qadri Hamarsheh."Lectures for Neural networks and Fuzzy logic" at Philadelphia University on 2015/2016.

[20] Simon Haykin."Neural networks a comprehensive Foundation second edition."McMaster University Hamilton, Ontario, Canada.